USB TO USB DATA TRANSFER WITHOUT PC

[1]AMBIKA R, [2]VINOD KUMAR K V, [3]SWAROOP K, [4]SACHIN K, [5]LAKSHMANA D

[1,2,3,4,5]Dept. of Electronics and Communication Engineering, BMS Institute of Technology, Bangalore, Karnataka, India

Abstract- The Computer has become an inseparable part of life for many people in the world. Its innumerable uses have changed the life of many, not only in technical field but also in non-technical field. Moreover, now-a-days, with increasing use of the computer, people are getting well aware of the peripherals that are associated with it. These peripherals give additional features to the already existing functionalities of the computer, while some enhance them. Along with the advancements in capabilities of the Computer, the capabilities of these peripherals have also been improved, thus, making it simpler and more user-friendly for the user. Gradually, using the computer has become very easy these days. To make it even more simple, the interfacing of the peripherals with the computer have also changed and improved a lot. The main idea of this project is related to one such peripheral, the USB. This user friendly device is actually recognized by its Connection type- the BUS; called the Universal Serial Bus. It comprises of just four connection wires between the Host (Computer) and the Device, governed by a set of rules called the Protocol. The aim of this project is to transfer data without using a computer, from one USB flash disk to another.

Keywords- USB, FAT, Flash disk, NTFS, VNCL1

I. INTRODUCTION

Numerous applications arise in the daily life of a common Computer user that he has to Transfer data from one Universal Serial Bus (USB) Flash device into another, that too very quickly and on the go. For this, he has to first find a computer, wait for it to boot up, then plug in his device and transfer the data. Different types and makes of USB flash devices are used now-a-days. It is not necessary that all of these devices are supported by the computer and the operating system and their device drivers are available and installed. Carrying a computer or a laptop justfor the sake of data transfer is not affordable these days, where people want all devices to be handy. Moreover, transferring data via a computer involves a lot of power to be wasted, since the computer has to be entirely functional before it can transfer data.

Also, the threat of viruses and malware has made the life of common computer users all the more complicated. These viruses get activated as soon as the device is plugged into the system and get copied along with other data from one flash device into another.
The device will have following features:

- Small, lightweight and handy device.
- Powered by 9V battery.
- Support for all USB Flash devices formatted with File Allocation Table (FAT32) file System.
- Support for USB 2.0.
- Plug and Go function.

II. LITRATURE SURVEY

A core team from Compaq, Hewlett Packard, Intel, Lucent, Microsoft, NEC and Philips leading the development of the USB Speciation, version 2.0, that has in-creased data throughput by a factor of 40. This backwards-compatible extension of the USB 1.1 speciation uses the same cables, connectors and software interfaces so the user will see no change in the usage model.

They will, however, benefit from an additional range of higher performance peripherals, such as video-conferencing cameras, next-generation scanners and printers, and fast storage devices, with the same ease-of-use features as today's USB peripherals.

- USB supports three bus speeds: high speed at 480Megabits/sec., full speed at 12 Megabits/sec., and low speed at 1.5Megabits/sec. The USB host controllers in recent PCs support all three speeds.
- The USB Host Controller is in charge of asking every USB device if it has any data to send. Because of this topology, Device cannot start sending data without being prompted.
- This justifies the need for a USB Host Controller and its Configuration.

III. DESIGN BASICS

A VNC1L Vinculum controller provides the interface between the LPC2138 as the system controller and a USB2.0 full speed port. This allows a USB flash memory drive connection.

Vinculum is specifically targeted at the embedded USB controller market and requires a minimum of external support components. One key feature of the Vinculum core is that its code length is significantly reduced compared with common MCU cores. Reducing the code overhead of the core allows much

more functionality to be squeezed into the on- chip e-flash memory.
Using Vinculum to link a small LPC2138 MP to a USB A and USB B connector and hence to a USB flash drive. Commands and data are sent via TXD (pin 6) to the VNC1L RXD (pin 32). VNC1L handles the FAT 12/16/32 file creation and data storage on the USB flash drive communicating with the drive via USB2DM and USB2DP on pins 28 and 29. Data is read from the flash drive via the same pins, and sent from the VNC1L TXD (pin 31) to the RXD (pin 5) of the LPC2138 for use by the system firmware.

The system is controlled by the firmware on the LPC2138, with the transfers controlled by instructions issued by the LPC2138 and interpreted by the standard firmware on the Vinculum. Whilst that is a simple description of the system, more is required to complete the design. The devices need power, crystals to control their clocks, and they need to be programmed. Using a 12MHz crystal on pins 61and 62 of the LPC2138 allows for higher baud rates of up to 9600bits/sec in its UART interface as opposed to the maximum 9600 bits/sec achievable by using the internal 8MHz oscillator, thus improving the performance of the system.

LPC2138 I/O pins10 (RXD) and 11(TXD) are used by their firmware to simulate RTS/CTS handshake signals with the VNC1Ls UART interface.

The VNC1L is shipped as a blank device. It must be programmed (when in boot loader mode) with firmware before use. It can be programmed either pre-assembly, using the VPROG-1 standalone programmer or programmed in-circuit via the UART interface. The VNC1L boot loader uses the UART interface to load new firmware into the Vinculum Flash memory. To enable the boot loader, the PROG pin must be driven low and VNC1L must then be reset by driving the RESET pin low then high. Run mode can be enabled by driving the PROG pin high and then resetting VNC1L by driving the RESET pin low then high.

When VNC1L firmware is updated via a microcontroller with a UART, the microcontroller must be capable of at least 115200 baud. The firmware can be upgraded in to the Flash memory via the UART interface or via a USB interface.

A 5V regulated PSU at 250mA is required, providing up to 200mA at the USB A connector, 25mA to power the VNC1L and 25mA to power the LPC2138 VNC1L requires a 3.3V supply which is provided by a 3.3V LDO regulator, and has 5V tolerant I/O pins which enable it to connect to the LPC2138 without using level shifters. For low-power applications the VNC1L can be put into a 2mA sleep mode when notrequired. To wake up the device, strobe the ring indicator pin (pin 16) of the UART interfaces. If this is connected to the RXD line, as here, it can be triggered by an incoming dummy command to wake up the device.

This design also includes a bi-color status LED indicator powered from pins 16 and 18. This indicates successful enumeration of the USB flash drive and access to the file system. The VNC1L is programmed with standard firmware, called VDAP (Vinculum Disk and Peripheral) that interprets the commands coming from the LPC2138. These VDAP commands are DOS-like instructions such as DIR, RD and WR. The command set also support single byte hex commands which are more suited to control by a microprocessor.

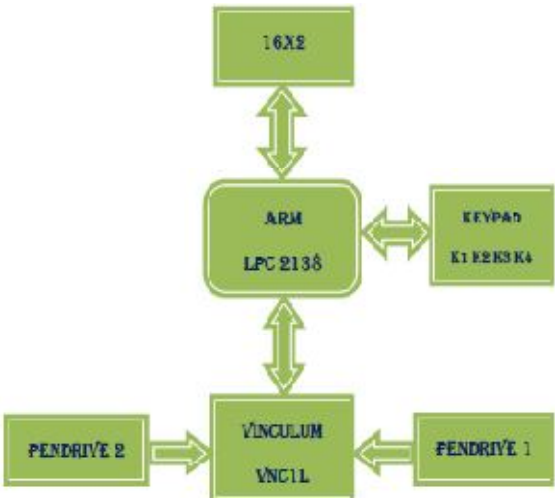

Fig. 1. Block Diagram

IV. RESULTS

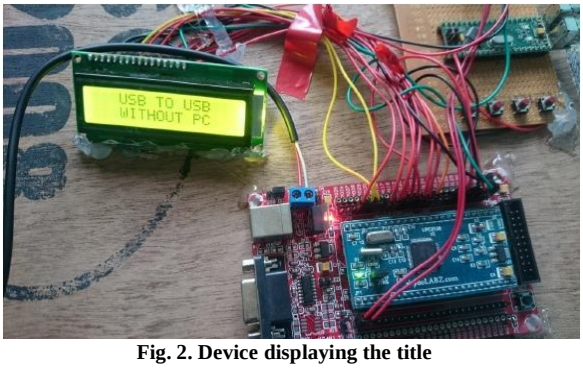

Fig. 2. Device displaying the title

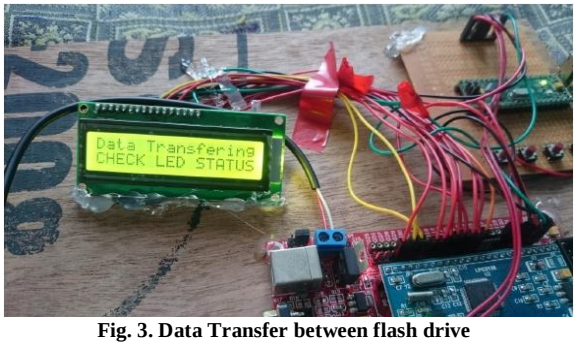

Fig. 3. Data Transfer between flash drive

CONCLUSION

USB to USB data transfer without connecting to PC allows a user to transfer data between two USB's when there is no PC available. It is user friendly and handy device.

This portable kit consumes less power. It also allows the data transfer between Bluetooth enabled device and USB device. Transferring the data through USB in today's scenario is the most common task. But the problem is that for transferring the data to a personal computer or laptop is difficult if u don't have any of them. It is affordable to purchase a USB data drive than purchasing a laptop or PC. Therefore we came up with a handled battery operated affordable device which can transfer the data between two USB data drives without the help of PC or laptop. The advantage of this device is that it is battery operated so there is no need of power supply connection and data transfer can take place at any place. Currently we have design the system for transfer of data up to 2GB only, but it can be increased by proper selection of the bus.

FUTURE SCOPE

While working on the development of the system and exploring the peripherals that can be interfaced with the ARM 7 we found that with little modification in the project several new features could be added. Following are the things that can be done with few modifications.

- Add USB host capability to embedded products.
- Interface USB Flash drive to MCU/PLD/FPGA.
- Using Bluetooth in our device, we can connect with any Bluetooth enable devices making the data transfer wireless.
- Keypad and Graphical LCD could be replaced by touch screens which could make human work easier by drag and drop method.

REFERENCES

[1]. JAN AXELSON, USB Complete, Penram Publications, 2nd edition, pp. 5-7. Nov1999.

[2]. JAN AXELSON, USB mass storage, Lakeview Publications, @2nd edition, chp.4,pp. 57-69. 2000.

[3]. John Hyde, USB design, a technical introduction to USB 2.0, white papers.

[4]. USB Implementers Forum Inc., www.usb.org/ interoductionusb-2.0papers/

[5]. www.alldatasheets.com

[6]. www.ftdichip.com/document Reference No.: FT-000006v. VinculumFirmware User Manual Version 2.05.

[7]. www.techpubs.sgi.com

[8]. USB Implementers Forum Inc., www.usb.org/ interoductionusb-2.0papers/

★★★